



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/714,273	11/17/2000	Govind Malalur	58268.09059	5515
32294	7590	08/01/2007		
SQUIRE, SANDERS & DEMPSEY L.L.P. 14TH FLOOR 8000 TOWERS CRESCENT TYSONS CORNER, VA 22182			EXAMINER AL HASHEMI, SANA A	
			ART UNIT 2164	PAPER NUMBER
			MAIL DATE 08/01/2007	DELIVERY MODE PAPER

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.



UNITED STATES PATENT AND TRADEMARK OFFICE

---

Commissioner for Patents  
United States Patent and Trademark Office  
P.O. Box 1450  
Alexandria, VA 22313-1450  
[www.uspto.gov](http://www.uspto.gov)

**BEFORE THE BOARD OF PATENT APPEALS  
AND INTERFERENCES**

Application Number: 09/714,273  
Filing Date: November 17, 2000  
Appellant(s): MALALUR, GOVIND

**MAILED**

**AUG 01 2007**

*Technology Center 2100*

---

Alicia M. Choi  
For Appellant

**EXAMINER'S ANSWER**

This is in response to the appeal brief filed March 28, 2007 appealing from the Office action mailed October 20<sup>th</sup>, 2006.

**(1) Real Party in Interest**

A statement identifying by name the real party in interest is contained in the brief.

**(2) Related Appeals and Interferences**

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

**(3) Status of Claims**

The statement of the status of claims contained in the brief is correct.

**(4) Status of Amendments After Final**

The appellant's statement of the status of amendments after final rejection contained in the brief is correct.

**(5) Summary of Claimed Subject Matter**

The summary of claimed subject matter contained in the brief is correct.

**(6) Grounds of Rejection to be Reviewed on Appeal**

The appellant's statement of the grounds of rejection to be reviewed on appeal is correct.

**(7) Claims Appendix**

The copy of the appealed claims contained in the Appendix to the brief is correct.

**(8) Evidence Relied Upon**

6,829,217

Bechtolsheim et al.

12-2004

**(9) Grounds of Rejection**

The following ground(s) of rejection are applicable to the appealed claims:

***Claim Rejections - 35 USC § 102***

(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

Claims 1, 2, 4-9, and 11-15 are rejected under 35 U.S.C. 102(e) as being anticipated by Bechtolsheim et al. (Bechtolsheim hereinafter) (US Patent No. 6,829,217 )

Regarding Claims 1, 8, and 15, Bechtolsheim discloses a method of performing a table look-up in a network device comprising the steps of:

receiving a data packet (Fig. 3, step 300 input packet Col. 5, lines 29-33, Bechtolsheim) through an input port of the network device (Col. 5, lines 4-12, Bechtolsheim);

parsing said data packet into an index portion and a corresponding bucket portion (Col. 5, lines 30-37, Bechtolsheim))

indexing, directly, said index portion to said corresponding bucket portion (Col. 6, lines 37-50, Bechtolsheim);

accessing address table information stored in an address look-up table (Col. 11, lines 55-60, Bechtolsheim);

Regarding Claims 2, and 9, Bechtolsheim discloses a method wherein said step of indexing said index portion to said bucket portion is the step of linearly indexing said index portion to said bucket portion (Col. 7, lines 43-47, Bechtolsheim).

Regarding Claims 3, and 10, Bechtolsheim discloses a method wherein said step of indexing said index portion to said bucket portion is the step of XOR indexing said index portion to said bucket portion (Col. 6, Table 2, lines 16-27, Bechtolsheim)

Regarding Claims 4, and 11, Bechtolsheim discloses a method further comprising the step of sorting said bucket portion (Col. 8, lines 37-43, Bechtolsheim).

Regarding Claims 5, and 12, Bechtolsheim discloses a method further comprising the step of binary sorting said bucket portion (Col. 8, lines 3-5, Bechtolsheim).

Regarding Claims 6, and 13, Bechtolsheim discloses a method wherein the step of parsing said data packet into an index portion and a corresponding bucket portion further comprises the step of parsing said index portion so that said index portion will recur when other data is parsed into said index portion and said corresponding bucket portion (Col. 8, lines 6-17, Bechtolsheim).

Regarding Claims 7, and 14, Bechtolsheim discloses a method further comprising the step of storing information regarding said data in said address look-up table as table information when no table information is available using said bucket portion to access table information (Col. 7, lines 17-25, Bechtolsheim).

#### **(10) Response to Argument**

Appellant argues, “ Bechtolsheim does not disclose any particular method, device or switch for parsing said data packet into an index portion and a corresponding bucket portion as recited in independent claim 1:

Examiner disagrees. The Bechtolsheim discloses a device as stated in for example in Col. 3, lines 61-63, and the teaching of parsing data is the method for indexing the data packet, in fact the Bechtolsheim discloses a method of parsing data in multiple ways, as shown in Fig. 3, step 300 the input packet which corresponds the “receiving data packet through an input port of the network device”, “parsing said data packet into an index portion and corresponding bucket portion” the applied art at Col. 5, lines 31-41, the packet header is parsed 302 to determine the packet size, source address, destination address, and type of service (TOS). Additionally, the UDP source and destination port (for an IP packet) or the MAC source and destination and protocol type (for Ethernet packets) may be extracted as required to fully identify the necessary TOS. Refer to FIG. 2 for the bitwise locations of this information within the industry-standard IP packet header 200. The number of buffer elements or cells, which may be counted in terms of bytes or groups of bytes, required to buffer the incoming packet is computed. This is one of the parsing method disclosed by the applied art the other method of parsing the data as shown in Col. 6, lines 1-26 In a substantially parallel process, the extracted header data is transformed by calculating a hash index 332 according to the following function, expressed in the C programming language:

```
(1)  hdr_ip* iph = (hdr_ip*)pkt .fwdarw. // get pointer to IP header
access(off_ip_ ( corresponds to the index portion); portion of packet int i = (int)iph .fwdarw.
src( ) (corresponds to the bucket portion); // get
source IP address int j = (int)iph .fwdarw. dst( ); // get destination IP
address int k = i + j; // add src and dst return (k + (k >> 8) + //
shift, add, divide modulo a -(k >> 4)) % ((2 << 19) - 1); large prime
```

Art Unit: 2164

(11) Alternatively, the following function can also be used to calculate hash index 332:

```
(2)  hdr_ip* iph = (hdr_ip*)pkt .fwdarw. // get pointer to IP header
access(off_ip_); portion of packet int i = (int)iph .fwdarw. src( ); // get
source IP address int j = (int)iph .fwdarw. dst( ); // get destination IP
address i = i j; // XOR src and dst i = i &gt;&gt; 16; // shift high order to
low order i = i &gt;&gt; 8; //shift again return i;
```

Appellant argues that the “Bechtolsheim does not teach or suggest that the same index is regenerated using a bucket portion”.

Examiner disagrees. The reference is required to disclose the teaching of indexing parsed data as cited in the claim language, appellant argument regarding regenerated using a bucket portion not in the claim language, therefore the office believe the indexing as disclosed in the applied art corresponds to the claimed indexing.

Appellant argues, “Bechtolsheim fails to teach or suggest all of the recitation of independent claim 1.

Examiner disagrees. And based on the responses stated above, the examiner believes all the limitations have been addressed.

Appellant argues regarding claim 2, “said step of indexing said index portion to said bucket portion is the step of linearly indexing said index portion to said bucket portion”. The cited portion of the applied art discloses a The flow identifying information contained in the packet header (sometimes called the "flow label" in the art) is hashed in order to reduce the huge range of packet header values into a single compact, easily manipulated field having a far smaller range of values. Hashing avoids the per-flow lookup, set up, and tear down overhead of

Art Unit: 2164

prior-art systems. For example, this embodiment of the present invention does not maintain flow table entries for each and every flow. Rather, of the  $2^{sup.160}$  possible flows uniquely identified by the first five 32-bit words in the IP packet header, the hash function limits the flow table to just  $2^{sup.n}$  entries, substantially less than the unhashed situation. In other words, flow table 335 consists of  $2^{sup.n}$  entries, where  $n$ =the number of bits in the output of the hash function above. In one embodiment, a 19 bit hash index is used, supporting 512K entries. This provides the advantage of needing fewer bits to identify a table entry corresponding to a particular flow, thus reducing the overhead and resource cost of this particular embodiment over the prior art. And The hash index is stored in the packet descriptor field in the transmit (output) queue for later use in transmitting the packet (FIG. 4). The packet descriptor field also contains the packet length, rewrite information, and a pointer to the start of the data buffer for the packet. Wherein the indexed portion is related to bucket portion as recited in claim 2.

Appellant argues regarding claim 3, the reference did not disclose the “said step of indexing said index portion to said bucket portion is the step of XOR indexing said index portion to said bucket portion”.

Examiner disagrees. The cited table on Col. 6, lines 20-26, discloses the claimed/argued limitation. `hdr_ip* iph = (hdr_ip*)pkt .fwdarw. // get pointer to IP header  
access(off_ip_); portion of packet int i = (int)iph .fwdarw. src( ); // get  
source IP address int j = (int)iph .fwdarw. dst( ); // get destination IP  
address i = i j; // XOR src and dst i = i &gt;&gt; 16; // shift high order to  
low order i = i &gt;&gt; 8; //shift again return i.`



Art Unit: 2164

**(11) Related Proceeding(s) Appendix**

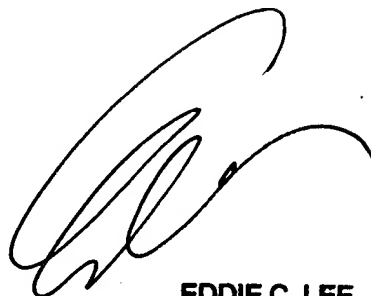
For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

/Sana AL-Hashemi/  
Primary Examiner  
Art Unit 2164


Conferees:

Eddie Lee  
Supervisory Patent Examiner



**EDDIE C. LEE**  
**SUPERVISORY PATENT EXAMINER**

Charles Rones  
Supervisory Patent Examiner



**CHARLES RONES**  
**SUPERVISORY PATENT EXAMINER**

Alicia M. Choi  
SQUIRE, SANERS & DEMPSEY LLP  
8000 Towers Crescent Drive, 14<sup>th</sup> Floor  
Tysones Corner, VA 22182-2700